Transferring deep knowledge for object recognition in Low-quality underwater videos

Transferring Deep Knowledge for Object Recognition in Low-quality Underwater Videos

Xin Sun[1,2], Junyu Shi[1], Lipeng Liu[1], Junyu Dong[1,*], Claudia Plant[3], Xinhua Wang[2], Huiyu Zhou[4]

[1] College of Information Science and Engineering, Ocean University of China, 266100 Qingdao, PR.China

[2] State Key Laboratory of Applied Optics, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, 130033 Changchun, PR.China

[3] Faculty of computer science, University of Vienna, Vienna, Austria

[4] School of Electronics, Electrical Engineering and Computer Science, Queen's University of Belfast, Belfast, United Kingdom

In recent years, underwater video technologies allow us to explore the ocean in scientific and noninvasive ways, such as environmental monitoring, marine ecology studies, and fisheries management. However the low-light and high-noise scenarios pose great challenges for the underwater image and video analysis. We here propose a CNN knowledge transfer framework for underwater object recognition and tackle the problem of extracting discriminative features from relatively low contrast images. Even with the insufficient training set, the transfer framework can well learn a recognition model for the special underwater object recognition task together with the help of data augmentation. For better identifying objects from an underwater video, a weighted probabilities decision mechanism is introduced to identify the object from a series of frames. The proposed framework can be implemented for real-time underwater object recognition on autonomous underwater vehicles and video monitoring systems. To verify the effectiveness of our method, experiments on a public dataset are carried out. The results show that the proposed method achieves promising results for underwater object recognition on both test image datasets and underwater videos.

1. Introduction

Despite the fact that the ocean plays a very foundational role of human life, we have a limited ability to explore the underwater world for a long time in history. Today's technologies and materials allow us to explore the ocean in deep and observe the undersea environment continuously. Undersea exploration can help us to better understand marine ecosystems and environmental changes. Autonomous underwater vehicles (AUV) and video monitoring systems give us opportunities to make detailed observations and collect samples of unexplored ecosystems. Specially underwater video techniques play an important role in observing macrofauna and habitat in marine ecosystems [1, 2], which provide abundant information for oceanography and fisheries science research. Underwater video based applications are increasingly developed in marine ecology studies and fisheries management. The most popular and widely reported cases in literatures are counting and measuring fish [3], investigating coastal biodiversity [1], observing species behavior [4], and exploring the undersea terrain [5].

Object detection and recognition techniques have been commonly used on videos analysis for the assessment of animal populations. With the underwater cameras, in recent years, a few research studies have been investigated for fish detection, recognition [6], tracking [7] and counting. In contrast to the conventional fishery monitoring approaches including mark-recapture techniques and gill netting [8], the underwater video based methods have advantages such as accurate species counting due to long term observation and environmental sustainability without disturbing their habitat. However, the low-light and high-noise scenarios pose several great challenges for the underwater video analysis. (1) Firstly, low illumination environments cause relatively low contrast background, which can confuse the traditional interest point detectors and produce weak descriptors. (2) Secondly, the object may appear to be of significantly different shapes over various camera angles due to the freely swimming environment. (3) Thirdly, most of underwater videos are of low resolution and low saturation, thus discriminative information is limited to recognize objects from the videos. Above all, most state-of-the-art image and video analysis methods suffer seriously from these drawbacks.

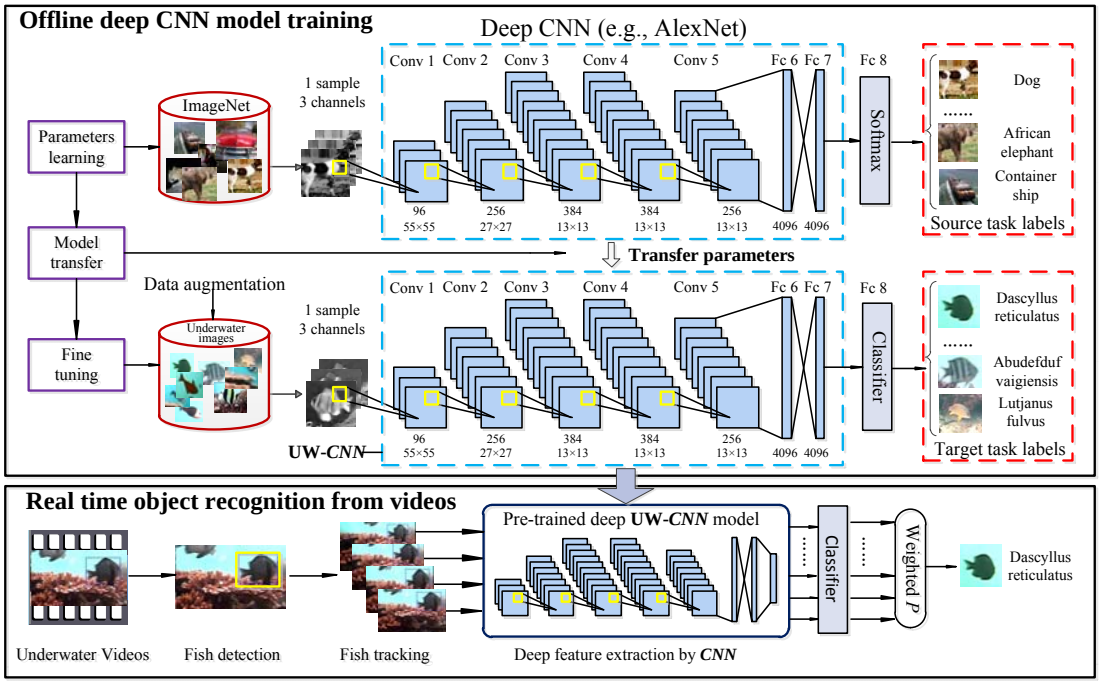


Figure 1. Illustration of the proposed framework by taking AlexNet as an example

All the above issues motivate us to design a novel solution for underwater object recognition from low-contrast and low-resolution underwater videos. Figure 1 shows the proposed framework for object recognition tasks on underwater videos. It can be seen from the illustration that an offline deep Convolutional Neural Network (CNN) model is firstly learned by proposing a transfer approach in order to overcome the insufficient training data problem. Then, with the pre-trained underwater *CNN* model (UW-*CNN*), a real time object recognition system is designed for underwater videos. The advantages of this work is that: (1)As the interesting points are difficult to be detected from the low-contrast and low-resolution images, the state of the art CNN method gives us a chance to produce abstract discriminative features from the object.

Figure 2 illustrates a comparison between the SIFT and CNN results. We can see that only a few interest points are detected on the object using the SIFT method. Most of them are tedious and do not contain powerful discriminative information. So it is better to identify the object from its shapes rather than local features. Figure 2 also visualizes part of the middle layers of the CNN output. It can be observed that the global shape information is well captured. (2)To overcome the predicament of "data-hungry" of CNNs with limited underwater training data, we introduce the transfer learning to learn a special CNN model for underwater object recognition together with the help of data augmentation tricks. The data augmentation simulates various possible shapes of the object from normal ones to improve the robustness of the CNN model. (3) To identify the object from videos, we consider the importance of the objects presented in the successive frames. For the final decision of object recognition, the object closer to the camera should have a higher weight than the others. So in the real time object recognition system, a weighted probabilities decision mechanism is used.

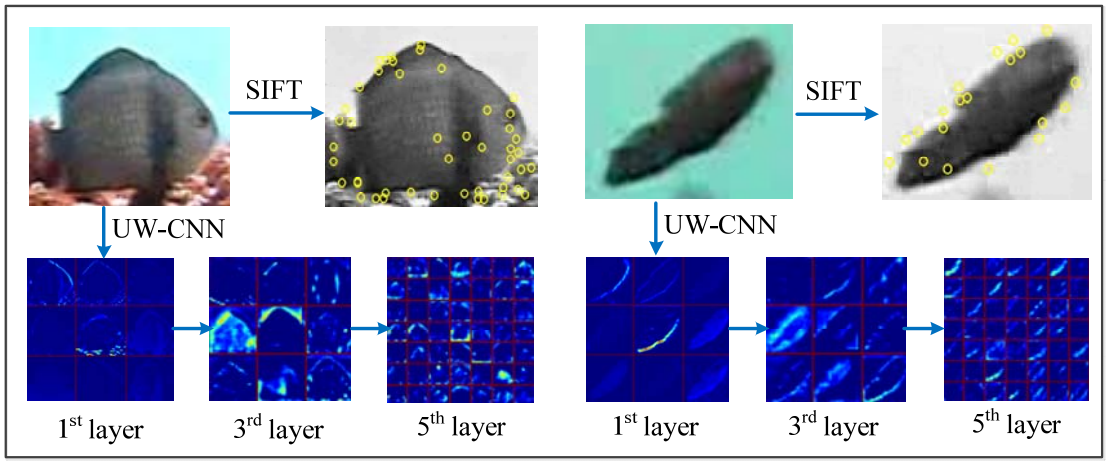


Figure 2. An comparison illustration between the SIFT and CNN

The main contributions of this work include:

- We use the deep CNN model [9] for underwater objects recognition from low-contrast and low-resolution underwater videos, which can better achieve illumination invariant and overcome the challenges caused by low quality videos.
- We overcome the difficulties imposed by small size underwater training data by proposing a transfer learning framework, which takes a fully-trained model from the ImageNet challenge as prior knowledge. Moreover we enlarge the training dataset by horizontal mirroring, rotating, subsampling and affine transformation in order to enrich the varieties of the image dataset. To the best of our knowledge, it is the first to use deep knowledge transferring method in the special field of underwater object recognition
- A weighted probabilities decision mechanism based on trajectory is applied to identifying objects. Then we propose a practical deep based application for underwater video analysis.

2. Related work and background concepts

This work is related to numerous works that have been reported in the fields of machine learning and computer vision, specifically in transfer learning and deep

learning. This section shortly reviews the related works, and presents some fundamental concepts needed for the understanding of this work.

2.1 Object detection in underwater videos

Marine biologists have employed the underwater video techniques in marine ecology studies for many years [10, 11]. To monitor a marine ecosystem, researchers have widely used computer vision techniques to detect underwater objects automatically [2, 3]. About thirty years ago, stereo photographic techniques have been applied for determining the size and relative position of free-swimming sharks [12]. Lines et al. [13] developed an image analysis based system for estimating the mass of swimming fish from video frames under a limited range of conditions. Harveya et al. [14] designed a stereo–video camera system to measure the accuracy and precision of the length and maximum body depth of tuna. An automated system was developed to detect and track objects in underwater videos collected by remotely operated underwater vehicles (ROVs) for oceanographic research [15]. Spampinato et al. [16] presented a machine vision system for detecting, tracking and counting fish from real-time videos, which consists of a series of video texture analysis, object detection and tracking procedures. Later they also proposed an automatic fish classification system to assist marine biologists in understanding species' behaviors in a natural underwater environment [17]. Hwang et al. [18] reported an automatic segmentation algorithm for fish acquired by a trawl-based underwater camera system. They overcame the low brightness contrast problem in the underwater environment by adopting a histogram back-projection procedure on double local-thresholded images. Typically, Fisher et al. [19] presented a research tool to support marine ecologists' research by allowing the analysis of long-term and continuous fish monitoring underwater videos. It is suitable for discovering ecological phenomena such as changes in fish abundance and species composition over time and different areas.

Lee et al. [20] widely investigated the vision-based object detection and tracking techniques for underwater robots. In their work, numerous approaches have been tested to overcome the limitations of underwater cameras, such as a color restoration algorithm for the degraded underwater images, detection and tracking methods for underwater target objects. Currently the biggest challenge for the underwater video analysis is the low-light and high-noise caused by the uncontrolled illumination and noisy video capturing environment. Chuang et al. [21] tried to overcome such difficulties and proposed a multiple fish-tracking algorithm for trawl-based underwater camera systems. Charalampidis et al. [22] also tried to solve the blurry and poor illumination problem and proposed a background subtraction and image segmentation method for images obtained using a two camera stereo system. However the above mentioned methods rely heavily on manmade discriminant features, which are hardly captured in low quality images. Accordingly this work resorts to an abstract feature extracting method, such as deep feature learning.

2.2 Deep Convolutional Neural Networks (CNN)

Great successful advancements of computer vision have been witnessed in the past several years due to the emerging technologies of deep learning and big data. Deep features extracted from CNN have achieved better performance than handcraft

features (e.g., LBP, SIFT etc.) by a significant margin in many vision tasks, such as ImageNet challenge [9]. Most of the researchers begin to tackle their vision problem using deep learning methods. The idea of the CNN was proposed nearly twenty years ago by LeCun [23], and achieved impressive performance with the GPU hardware deployment in recent years.

A CNN is an architecture formed by a stack of convolutional and fully connected layers where the output of one layer is the input of the following layer, and essentially differs from other neural networks by incorporating local connections, weight sharing and local pooling. A well-known CNN model is AlexNet which is first introduced for the image classification challenge by Krizhevsky [9]. By visualizing the features of each layer, Zeiler and Fergus [24] found that the first layer of the network usually learn low-level features such as edges and corners, and further layers learn high-level features. Currently the CNNs model has been widely used as a powerful discriminant tools for object detection and recognition [25, 26], and obtained state-of-the-art results in many different applications. A typical research work [27] about underwater live fish recognition based on deep architecture is reported based on two convolutional layers. They used a spatial pyramid pooling (SPP) to extract information in variant to large poses.

Some of the researches have shown that CNN models trained using the ImageNet can be regarded as generalized feature extractors, which powerful high level features are produced for many new related tasks [28, 29]. The following section will give a short introduction about how to transfer the well trained model to new tasks.

2.3 Transfer learning

A major hypothesis for most machine learning tasks is that the training and future data must have the same distribution and be in the same feature space [30]. However, in many current real world applications, we cannot obtain sufficient training data in some domain-specific tasks; meanwhile abundant training data are available in the general domain. The study of transfer learning is focused on repurposing the knowledge learned previously to solve new problems with better solutions. This section briefly introduces the definitions of transfer learning and more detailed reviews can be found in literature [30].

Definition of transfer learning [30]. *Given a source domain $\mathcal{D}_S=\{\mathcal{X}_S, P(X_S)\}$ and learning task $\mathcal{T}_S=(Y_S, f_S)$, a target domain $\mathcal{D}_T=\{\mathcal{X}_T, P(X_T)\}$ and learning task $\mathcal{T}_T=(Y_T, f_T)$, transfer learning aims to help improve the learning of the target predictive function f_T in $\mathcal{D}_T$ using the knowledge in $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S\neq\mathcal{D}_T$, or $\mathcal{T}_S\neq\mathcal{T}_T$. The condition $\mathcal{D}_S\neq\mathcal{D}_T$ implies that either $\mathcal{X}_S\neq\mathcal{X}_T$ or $P(X_S)\neq P(X_T)$. And the condition $\mathcal{T}_S\neq\mathcal{T}_T$ implies that either $Y_S\neq Y_T$ or $f_S\neq f_T$.*

When the target application creates huge data, but lacks of groundtruth data to train a deep network model from scratch, transfer learning can be a powerful tool to enable training a large target network without overfitting [31]. Deep features extracted from CNNs trained on large annotated ImageNet dataset have been successfully used as generic features for various new vision tasks [28, 31], e.g. arduous recognition task [32, 33]. In this work, transfer learning is introduced for the object recognition task in low-contrast and low-resolution underwater videos. It takes the full advantage of

remarkable discriminative power and well trained hyper-parameters of the deep CNNs.

3. Real time object recognition framework

3.1 Offline deep CNN model via transfer learning

A key advantage of Deep Learning is its ability of learning stable and robust features from massive amounts of data. However one of the most important preconditions is that an informative dataset should be collected, which is used to estimate millions of parameters used by deep layers. It is difficult and costly to obtain an ideal largely labeled underwater image data. To solve this problem, this work firstly enlarged the dataset using label-preserving transformations [34], then proposed a transfer learning approach for the underwater object recognition task with insufficient underwater images. The trained deep CNN model can be then used for real time live object recognition from underwater videos.

A. Data augmentation

For illustrating the procedure of proposed system, this paper takes the ground-truth data from the Fish4Knowledge project [19] to train the deep UW-*CNN* model for underwater object recognition. The fish species are manually labeled by following the instructions from marine biologists [35]. To enhance the robustness of our model, we extend the image dataset with extra three categories, i.e., stone, coral and seawater. We will show the necessity and practicability in the later section.

A large amount of data is one of the core issues for deep learning, making the learning model effective and preventing overfitting. As the size of training underwater image data is small, data augmentation can be one of keys to improve the model performance. In this work, we employ four distinct forms of image set augmentation, i.e., horizontal mirroring, crop, subsampling and affine transformation, all of which allow transformed images to be generated from the original images with the same label.

We first simply flip all the images horizontally to simulate fishes swimming in different directions. To simulate the environment under occlusion, we then crop the images by removing about one third on the right and left respectively. Thirdly, as the objects can present in any distance in the front of the cameras, we downsample the images to make the objects seem a little far away from the cameras. Finally a sine transform based affine transformation is applied to each training image so as to produce more images of every object from different view angles.

In many imaging systems, affine transformation is applied to tackle pose-invariant object recognition problem by geometrically warping the specific pose into the frontal pose [36]. To the contrary application in our work, affine transformation is used to warping the frontal pose of objects to form various views. The affine transformation can be equally regarded as the composed effects of translation, rotation, isotropic scaling and shear. For an object image, we first insert it into a plane coordinate system with the center of image aligned at the origin of the coordinate system. Then different scaling transforms act on the images according to the coordinates. Given an $m \times n$ image I, and the coordinates of each pixel is denoted as $\{(x,y) \in I \mid x=1,2,\ldots,m; y=1,2,\ldots,n\}$, the new value of x coordinate can be calculated as formula (1) by affine

transformation with angle α.

$$x' = \begin{cases} x + sin\,\alpha\left(x - \frac{m}{2}\right), & x \geq \frac{m}{2} \\ x - sin\,\alpha\left(x - \frac{m}{2}\right), & x < \frac{m}{2} \end{cases} \quad (1)$$

The formula (1) shows that affine transformation act on the image according to the position of x. And the new image is formed as $\{(x', y) \in I \mid x'=1,2,\ldots,m; y=1,2,\ldots,n\}$. The image dataset is quadrupled in size by affine transformed with $\alpha=-10°, -20°, 10°$ and $20°$. It can be seen that the affine transformation is only performed in horizontal. The reason is based on the observation that underwater objects (especially for fishes) always keep their body vertical in the water.

B. Transfer learning for low-contrast and low-resolution underwater images

As mentioned before, more than 60 million parameters contained in the CNN architecture have to be learned with a large amount of training data. Given the "data-hungry" nature of CNNs and the difficulty of collecting large-scale underwater image datasets [28], the applicability of CNNs directly to our underwater object recognition tasks appears a quite difficult challenge. To address this problem, we propose a transfer learning approach for the special object recognition task by transferring the parameters from a full trained model and fine-tuning the model using a limited amount of underwater images.

The ImageNet dataset is a complicated benchmark in object category classification and object detection. It consists of 14 million images with 1000 categories [37]. Our solution of underwater object recognition is to transfer knowledge from the source CNN model learned via ImageNet to our target domain, as illustrated in Figure 1. However the labels, quality and distribution of images are quite different in the source and target domains. The CNN model learned from ImageNet cannot be directly used as a feature extractor in the underwater environment. So the key idea of our solution is that the knowledge from the source domain will be recognized as priori values for the parameters of the target CNN model. To achieve our goal, we have to design a same architecture of CNN in the source and target domains. Then the parameters of the CNN model in the source domain (here ImageNet) are transferred as the initialization of the target CNN model. Finally the training procedure in the target domain could be taken as supervised fine-tuning task with our augmented underwater dataset to seek a suitable model. This work can also be regarded as transferring the recognition capabilities from general domain to a specific domain.

We formalize the transfer problem as follows.

(1) The source domain $\mathcal{D}_S=\{\mathcal{X}_S, P(X_S)\}$ is the ImageNet learning problem, where X_S is the learning sample of ImageNet and $\mathcal{X}_S$ is the feature space output from CNN. The task $\mathcal{T}_S=(Y_S, f_S)$ of ImageNet classification with deep CNN consists of two components: a label space Y and an objective predictive function f_S. Here the function f_S is the deep CNN model, and $\mathcal{K}_S$ is the parameters contained in the first seven layers which can be learned from the training data.

(2) The target domain $\mathcal{D}_T=\{\mathcal{X}_T, P(X_T)\}$ is the underwater object recognition problem. The learning task $\mathcal{T}_T=(Y_T, f_T)$ is to train the function f_T with the help of the

source domain. In our transfer learning task, the source and target domains are different, i.e., $X_S \neq X_T$ and $Y_S \neq Y_T$. The aim of the transfer task is to improve the target function f_T using $\mathcal{K}_S$ from the source task as priori knowledge. Our method is a parameter-transfer approach based on the hypothesis that individual models for related tasks should share some parameters or prior distributions of hyperparameters [30].

C. Network architecture and knowledge utilization for object classification

In order to illustrate how the framework works, we take the AlexNet model [9] as an example. We first employ the eight layers deep AlexNet model [9] trained on ImageNet as the source CNN model. The architecture of AlexNet, as shown in Figure 1, has five successive convolutional layers (*C*1…*C*5) and two fully-connected layers (*FC*6, *FC*7). An additional softmax classification layer, also called fully-connected *FC*8, is added to the end of the network to predict the scores for the 1000 categories. More specifically, details of the description of the geometry of the seven convolutional layers and their setup regarding contrast normalization and pooling can be found in the literature [9]. As shown in Figure 1, our solution only transfers the seven layers (*C*1…*C*5, *FC*6 and *FC*7) of the source domain to the target. As the target and the source tasks contain totally different categories, the classification layer *FC*8 should be trained separately and can be removed by strong classifiers.

On the source task, the weights in each layer are initialized from a zero-mean Gaussian distribution with a standard deviation 0.01; the biases in the 2nd, 4th, 5th, and fully-connected layers are initialized with the constant 1, others with 0. The parameters of layers *C*1…*C*5, *FC*6 and *FC*7 are trained on ImageNet. More details about the training can be found in the literature [9]. We will finally get a well-trained deep CNN model named ImageNet-CNN, and the knowledge $\mathcal{K}_S$ from this model. Here the knowledge $\mathcal{K}_S$ is the well estimated parameters of the first seven layers.

On the target task, we will train a CNN model with the same architecture of seven layers *C*1…*C*5, *FC*6 and *FC*7 as the source task. The parameters of the target model are initialized by the knowledge $\mathcal{K}_S$. As the first seven layers initialized by the knowledge $\mathcal{K}_S$ is already a general extractor, the training procedure in the target task is indeed a fine-tuning problem which makes the parameters suitable for the special domain. So we should set much smaller initial learning rates (e.g., use 0.001 instead of 0.01) for the first seven layers and fewer iterations in the training procedure. Furthermore the classification function in the source CNN model is a softmax classifier that obtains the scores of 1,000 categories of the ImageNet. For the target CNN model, we initialize a new classifier with random values of 26 categories (in case of Fish4Knowledge dataset). Here the parameters of the *FC*8 should be learned only from the underwater image set. So the learning rates for the eighth layer can be set as same as the source task, or even faster due to its fewer iterations.

Finally we get a new deep CNN model called UW-*CNN* as shown in our illustration example of figure 1, which can be used as a classifier in the following underwater vision system.

3.2 Real time object recognition from videos

Based on the deep CNN model described above, this work further develops a vision system for real time live object recognition from underwater videos.

A. Object detection and classification

Detecting objects from the underwater videos is the first step of the system. This can be done by resorting to background modeling approaches, which are common used for detecting the moving objects in the scene like in video surveillance [38]. The key idea is to build a model of the background and compare this model with the each frame in order to detect objects where a significant difference occurs. While a static background model might be appropriate for analyzing videos in well constrained environments, it is ineffective for most practical situations such as unconstrained underwater videos. The underwater videos are really unconstrained videos suffering from some technical issues including sudden light changes, low-quality videos, and bad weather conditions [39, 40]. This work employs the ViBe [41] background subtraction method as the object detection algorithm, which is a fast and robust non-parametric model [42]. Even the ViBe method may pose some false positive patches such as stones and corals, such problem can be well handled by our UW-*CNN* classification model.

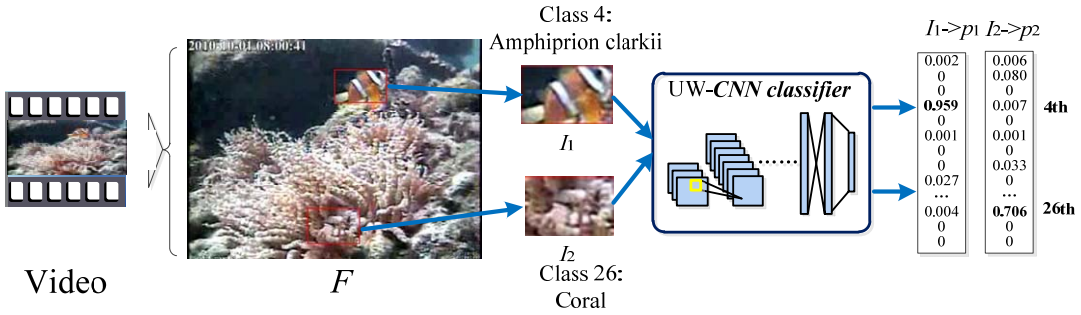


Figure 3. An illustration of the object classification procedure for one frame

Give one underwater video as shown in figure 3, the object detection module will produce a series of patch proposals $F\{I_1, I_2,\ldots, I_n\}$ for each frame F by ViBe. Every patch will be the input of the classifier UW-*CNN* and obtains a label distribution vector. The one achieved the highest probability is regarded as the label of these patches, such as the label Amphiprion clarkia for the first patch. Figure 2 also shows that the classifier compensates for the mistake (unwanted-object patch: coral) made by ViBe-based object detection due to the extension for the categories of the dataset.

B. Trajectory based object recognition

Individually identifying objects frame by frame has some disadvantages and is meaningless for practical applications. For example, (1) the classification accuracies are decreasing as the object moving far away from the cameras, which may cause a lot of misclassification results for the same object; (2) one object may exhibit various postures as they are freely swimming and switching directions very often, which may also confuse the classifier; (3) it is more practical that statistical records should be done based on videos instead of frame such as fish populations investigation and marine ecosystems monitoring. Thus it is better to implement the object recognition element based on a series of frames according to the motion trajectory of the object. For object tracking, we employ the covariance based tracking algorithm [43], which has been well proved in practice, to explore a series of patches $\{f_1, f_2, \ldots, f_T\}$ for the same fish in the video sequences.

This work presents a weighted probabilities decision mechanism which can be

applied on the trajectory to identify the objects. It can be formalized as formula (2), in which p_t is the label distribution given by the classifier for patch f_t, ω_t is the weight of f_t and T is the number of frames the object present. And the weight ω_t is defined based on the diagonal length of the image. The weights of each patch are defined by the normalized diagonal length of the image patch.

$$\text{label} = \text{argmax}_i P(i), \quad P = \sum_{t=1}^{T} \omega_t p_t \tag{2}$$

Figure 4 further gives an illustration for the calculation of label probabilities and its procedure of decision making. The object shows quite different orientations and appearances in its moving trajectory. When a 'dascyllus reticulatus' swims directly to the camera in the first frame, and is misclassified as 'scaridae' with the highest probability *0.620*. As it moves close to the camera and shows its clear outline, the classifier can predict a higher probability for the right category. It is evident that the closer it appears the more confident result we get. So we deduce a weight ω_t for each patch according to its size (the length of diagonal can better reflect the size of the image patch). It can be seen from Figure4, the output of the last patch will play a more important role in forming the final label distribution P with a higher weight. The final decision will be made according to the highest probability in the weighted-sum distribution P.

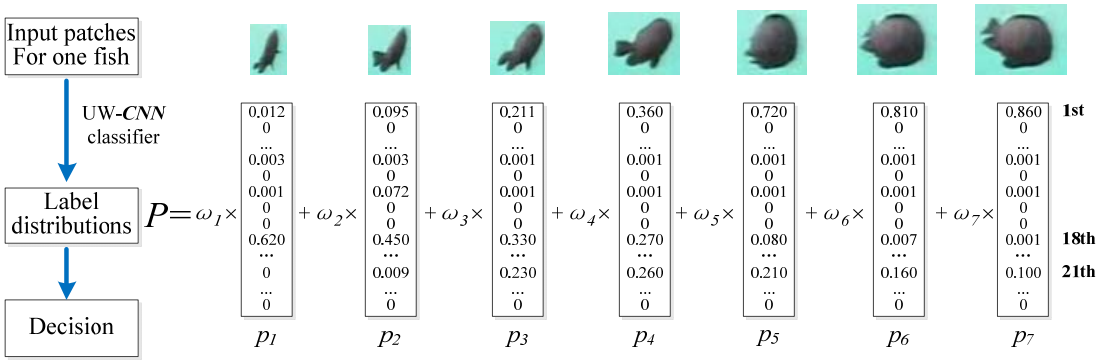


Figure 4. An illustration of the trajectory based fish recognition

4. Experiments and results

Simulations of the proposed system are carried out on the Fish4Knowledge video and the fish analysis dataset [19, 39]. The proposed UW-*CNN* model is evaluated on the 23 categories fish recognition ground-truth dataset [44]. The dataset is created from a live video dataset resulting in 27370 verified fish images. The whole dataset is divided into 23 categories and each category is presented by a representative species, which is based on the synapomorphies characteristic from the extent that the taxon is monophyletic. As mentioned earlier, we extend the image dataset with another three unwanted categories, i.e., stone (120), coral (93) and seawater (51), as the aid for false positive results by fish detection.

We use the Caffe [45] package to train and fine-tune the CNN with the same structure and parameter settings as Krizhevsky et al.[37] suggested. The codes for object detection and tracking are implemented using the OpenCV platform. And we ran experiments on a dual 8-core Intel Xeon E5-2650 processor, with a Tesla K40 (2880 cores and 12 GB of RAM).

A. Results of fish species recognition from images

Table 1 shows the distribution of the samples used in our experiments. The dataset is very imbalanced where the most frequent species is about 500 times more than the least one.

Table 1. Distribution of categories in the dataset

No.	Categories	Samples	No	Categories	Samples
1	Dascyllus reticulatus	12112	14	Zebrasoma scopas	90
2	Plectroglyphidodon dickii	2683	15	Hemigymnus	42
3	Chromis chrysura	3593	16	Lutjanus fulvus	206
4	Amphiprion clarkii	4049	17	Scolopsis bilineata	49
5	Chaetodon lunulatus	2534	18	Scaridae	56
6	Chaetodon trifascialis	190	19	Pempheris	29
7	Myripristis kuntee	450	20	Zanclus cornutus	21
8	Acanthurus nigrofuscus	218	21	Neoglyphidodon	16
9	Hemigymnus fasciatus	241	22	Balistapus undulatus	41
10	Neoniphon sammara	299	23	Siganus fuscescens	25
11	Abudefduf vaigiensis	98	24	Stone	120
12	Canthigaster valentini	147	25	Coral	93
13	Pomacentrus moluccensis	181	26	Seawater	51

For estimating the performance of the proposed method, the total images are divided into three subsets: 5/7 for training, 1/7 for validation, and 1/7 for test [27]. Actually it is a seven-fold cross-validation is used. Finally the average performance of the ten times running will be calculated as the final result.

As the proposed framework is general for employing any deep CNN models, we first compare the performance of various kind of existed CNN architectures, i.e., AlexNet model [9], GoogLeNet [46] and OxfordNet(VGG-16) [47]. The results are shown in Table 2. We can see that AlexNet achieves the best performance. AlexNet was originally designed to classify over ImageNet. OxfordNet and GoogLeNet was designed to be a direct improvement over AlexNet for the task of classifying ImageNet. Compared to AlexNet of 8 layers, they are more complex. For example, GoogLeNet has 22 layers, and need more computing power than Alexnet. So it is still difficult to fine-tune the transferred parametrical knowledge of OxfordNet and GoogLeNet with limited underwater dataset.

Moreover the input style of color model may also affect the performance. So we also carry out comparison experiments with different color models. Table 2 report the results with RGB and HSI image data as input respectively. It shows that the input of RGB images performs much better. The reason is that the Blue (B) and Green (G) channels can provide much more discriminative information because the blue-green channel is absorbed the least in water. Meanwhile the low illumination confuses the hue (H), saturation (S), intensity (I) channels.

Table 2. Results of classification precision with different deep networks via cross validation test

No	Categories	RGB-Alex	HSI-Alex	RGB -GLeNet	HSI -GLeNet	RGB-VGG	HSI -VGG
1	Dascyllus reticulatus	100.0%	99.79%	100.0%	60.00%	100.0%	0
2	Plectroglyphidodon	99.77%	99.87%	98.67%	99.61%	99.45%	99.70%
3	Chromis chrysura	99.60%	99.21%	99.21%	96.31%	99.79%	96.23%

4	Amphiprion clarkii	100.0%	100.0%	99.54%	98.52%	98.92%	99.16%
5	Chaetodon lunulatus	100.0%	100.0%	99.95%	99.78%	99.97%	99.92%
6	Chaetodon trifascialis	99.38%	99.38%	99.96%	99.82%	100.0%	99.82%
7	Myripristis kuntee	100.0%	99.46%	94.15%	93.57%	99.42%	94.15%
8	Acanthurus nigrofuscus	96.41%	95.41%	99.75%	76.30%	100.0%	98.27%
9	Hemigymnus fasciatus	100.0%	100.0%	84.18%	82.14%	92.35%	91.33%
10	Neoniphon sammara	100.0%	97.61%	98.61%	96.30%	99.54%	96.76%
11	Abudefduf vaigiensis	100.0%	100.0%	95.17%	99.63%	97.40%	99.63%
12	Canthigaster valentini	100.0%	100.0%	96.59%	84.09%	100.0%	90.91%
13	Pomacentrus	100.0%	98.21%	91.67%	95.45%	93.94%	96.21%
14	Zebrasoma scopas	100.0%	100.0%	100.0%	98.76%	100.0%	97.53%
15	Hemigymnus	100.0%	100.0%	75.31%	59.26%	95.06%	76.54%
16	Lutjanus fulvus	100.0%	100.0%	86.49%	83.78%	97.30%	89.19%
17	Scolopsis bilineata	100.0%	100.0%	99.46%	100.0%	98.92%	99.46%
18	Scaridae	96.56%	97.73%	88.64%	93.18%	97.73%	95.45%
19	Pempheris vanicolensis	100.0%	98.30.0%	100.0%	94.00%	100.0%	94.00%
20	Zanclus cornutus	100.0%	100.0%	100.0%	96.15%	100.0%	96.15%
21	Neoglyphidodon	100.0%	100.0%	72.22%	83.33%	94.44%	100.0%
22	Balistapus undulatus	100.0%	100.0%	71.43%	35.71%	85.71%	35.71%
23	Siganus fuscescens	100.0%	100.0%	88.89%	88.89%	94.44%	88.89%
24	Stone	100.0%	100.0%	90.91%	27.27%	100.0%	95.45%
25	Coral	100.0%	100.0%	77.78%	22.22%	100.0%	22.22%
26	Seawater	100.0%	80.00%	41.67%	0	66.67%	0
	Avg.	**99.68%**	98.67%	90.39%	79.39%	96.58%	82.80%

Hereby we take the RGB and AlexNet (RGB-Alex) as the input and CNN model of the proposed framework in the following experiments. Then, we compare four CNN-based method used as feature extractors against the representations for low-contrast and low-resolution underwater images classification. As mentioned in the above sections, a strategy of fine tuning all eight layers, including the eighth classifier layer, is suggested. According to the chosen classifier Softmax and SVM, we denote the UW-CNN model as CNN-Soft and CNN-SVM respectively. Another strategy commonly used is that only the last layer is fine-tuned by fixing the pre-trained seven layers, which is denoted as CNN-Last. We also try to train the CNN model direct with the underwater images as a baseline for transfer learning without data augmentation, which is named as CNN-Dir. We also report the classification precision results of DeepFish as a column name “DeepFish” [27].

The effectiveness of a method can be simply and directly measured by the classification performance on the dataset for classifiers. To better illustrate that, we report the precision and recall for every category as shown in Table 3 and Table 4 respectively. Precision is the fraction of the detected objects that belong to the correct category. Recall is the fraction of the objects that belong to the query category that are successfully retrieved [48].

Table 3. Results of classification precision with state of the art methods via cross validation test

No.	Categories	CNN-SVM	CNN-Soft	CNN-Last	DeepFish[27]	CNN-Dir
1	Dascyllus reticulatus	100.0%	99.78%	97.56%	99.31%	95.12%

2	Plectroglyphidodon	99.77%	98.79%	92.13%	97.13%	41.32%
3	Chromis chrysura	99.60%	99.75%	73.97%	98.64%	81.42%
4	Amphiprion clarkii	100.0%	99.97%	99.81%	100.0%	92.44%
5	Chaetodon lunulatus	100.0%	100.0%	99.38%	100.0%	95.15%
6	Chaetodon trifascialis	99.38%	100.0%	99.21%	92.59%	52.83%
7	Myripristis kuntee	100.0%	100.0%	95.58%	98.44%	84.55%
8	Acanthurus nigrofuscus	96.41%	89.05%	66.49%	64.52%	11.81%
9	Hemigymnus fasciatus	100.0%	98.15%	96.74%	100.0%	62.03%
10	Neoniphon sammara	100.0%	100.0%	100.0%	100.0%	100.00%
11	Abudefduf vaigiensis	100.0%	100.0%	100.0%	92.86%	63.16%
12	Canthigaster valentini	100.0%	100.0%	99.07%	95.24%	43.75%
13	Pomacentrus moluccensis	100.0%	96.09%	85.86%	100.0%	48.95%
14	Zebrasoma scopas	100.0%	85.06%	72.31%	84.62%	8.12%
15	Hemigymnus melapterus	100.0%	100.0%	91.11%	66.67%	47.37%
16	Lutjanus fulvus	100.0%	100.0%	99.47%	96.55%	0
17	Scolopsis bilineata	100.0%	100.0%	100.0%	85.71%	14.29%
18	Scaridae	96.56%	86.67%	40.94%	100.0%	33.33%
19	Pempheris vanicolensis	100.0%	100.0%	100.0%	100.0%	13.89%
20	Zanclus cornutus	100.0%	100.0%	100.0%	66.67%	33.33%
21	Neoglyphidodon	100.0%	84.62%	80.00%	50.00%	85.71%
22	Balistapus undulatus	100.0%	95.45%	68.85%	83.33%	8.03%
23	Siganus fuscescens	100.0%	100.0%	100.0%	100.0%	0
24	Stone	100.0%	100.00%	100.0%	null	null
25	Coral	100.0%	88.89%	80.00%	null	null
26	Seawater	100.0%	100.0%	100,0%	null	null
	Avg.	**99.68%**	97.10%	89.50%	90.10%	48.55%

For the sake of impartiality, we calculate the mean value of performance of methods with the first twenty-three categories. As can be seen from Tables 3 and 4, the average value of performance (in the "Avg." row) denotes that CNN-SVM performs better than the others.

Table 4. Result of recall via cross validation test

No.	Categories	CNN-SVM	CNN-Soft	CNN-Last	CNN-Dir
1	Dascyllus reticulatus	99.80%	99.40%	89.11%	73.18%
2	Plectroglyphidodon dickii	99.96%	99.96%	94.11%	97.89%
3	Chromis chrysura	99.66%	99.29%	94.80%	58.12%
4	Amphiprion clarkii	99.97%	99.95%	97.99%	91.19%
5	Chaetodon lunulatus	100.0%	99.91%	99.51%	86.59%
6	Chaetodon trifascialis	100.0%	97.48%	79.25%	52.83%
7	Myripristis kuntee	100.0%	98.24%	97.98%	52.39%
8	Acanthurus nigrofuscus	94.95%	94.44%	64.14%	15.15%
9	Hemigymnus fasciatus	100.0%	100.0%	98.11%	54.72%
10	Neoniphon sammara	100.0%	100.0%	100.0%	2.86%
11	Abudefduf vaigiensis	100.0%	95.56%	82.22%	13.33%
12	Canthigaster valentini	100.0%	100.0%	89.17%	29.17%
13	Pomacentrus moluccensis	100.0%	100.0%	98.84%	94.77%
14	Zebrasoma scopas	97.33%	98.67%	62.67%	70.67%
15	Hemigymnus melapterus	97.83%	100.0%	89.13%	19.57%

16	Lutjanus fulvus	100.0%	100.0%	97.40%	0
17	Scolopsis bilineata	97.96%	95.92%	95.92%	2.04%
18	Scaridae	100.0%	100.0%	100.0%	26.92%
19	Pempheris vanicolensis	100.0%	100.0%	94.74%	78.95%
20	Zanclus cornutus	100.0%	100.0%	100.0%	23.08%
21	Neoglyphidodon nigroris	100.0%	100.0%	72.73%	27.27%
22	Balistapus undulatus	100.0%	100.0%	100.0%	83.33%
23	Siganus fuscescens	100.0%	100.0%	100.0%	0
24	Stone	100.0%	90.91%	72.73%	null
25	Coral	100.0%	100.00%	100.0%	null
26	Seawater	100.0%	100.0%	100.0%	null
	Avg.	**99.45%**	99.08%	91.21%	45.83%

There are several kinds of transfer strategies for a pre-trained network being used as a feature extractor for these images. CNN-Last takes the pre-trained network on ImageNet and removes the last fully-connected layer (the classifier layer). Then it transfers the rest of the network as a fixed feature extractor and only retrains the layer to the new task. The results have shown that CNN-Last still performs better than CNN-Dir which is trained totally by the underwater images, however, much lower than the first two methods. The strategy as used in the first two methods is that transferring the super-parameters as initials and retraining the whole network with the limited underwater images. As shown in the results, the latter strategy is suitable for the situation where two domains are quite different from each other. And it can be seen that our source domain and target domain are quite different, e.g., high-quality images of the source and low-quality of the target. To further investigate the effective of the transferred knowledge of ImageNet domain, we transfer the learned deep knowledge at different layers from the well learned source model, and fine tuning the network with data augmentation. Detailedly, the first experiment is carried out without transfer any knowledge from the source domain. The second experiment is conducted by transferring the deep knowledge of the first layer from the source domain as initial parameters. And third experiment is employed deep knowledge of the first two layers from the source domain as prior knowledge. In order, the eighth experiment is transferring all the super-parameters as initials, which is the same as the above CNN-SVM. The comparison results are shown in Figure 5. We can see that the prior knowledge from source domain improves the performance of the special target task. Moreover, we can also give another conclusion by combining the result of CNN-Dir from Table 3. The data augmentation procedure, including horizontal mirroring, crop, subsampling and affine transformation, improves the performance from 48.55% to 61.54%. It shows that our solution works well for the nonrigid object deformation problem of underwater animals.

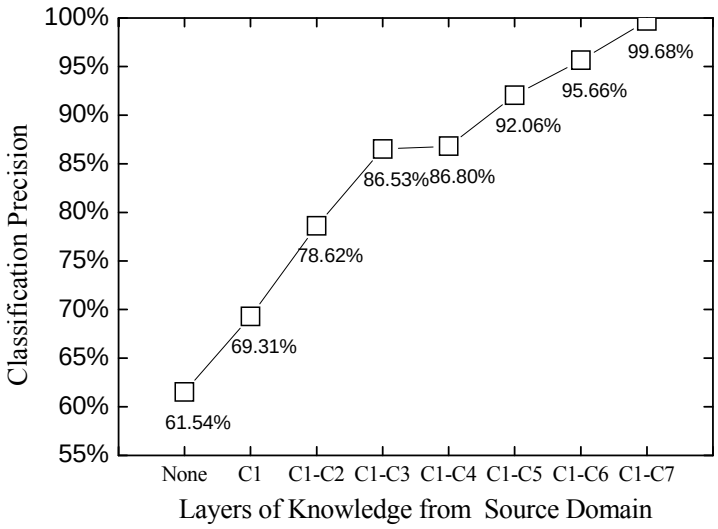


Figure 5. Results of classification precision with different layers of prior knowledge

At last, we take the Gabor features, Dense SIFT features and LDA features as traditional comparison methods. And the dense sift features of images are encoded as a fisher vector for each image. Then we train the SVM classifier with all the traditional features. Classification results are shown in the Table 5. It can be seen that the deep features achieve much higher performance than the manmade features. And our method with Alex as the network architecture achieves the best accuracy.

Table 5. Comparison results among deep and traditional methods

Method	Precision
Gabor	58.55%
Dsift-Fisher	83.37%
LDA	80.14%
DeepFish[27]	90.10%
RGB-Alex-SVM	**99.68%**

B. Real time live object recognition from videos

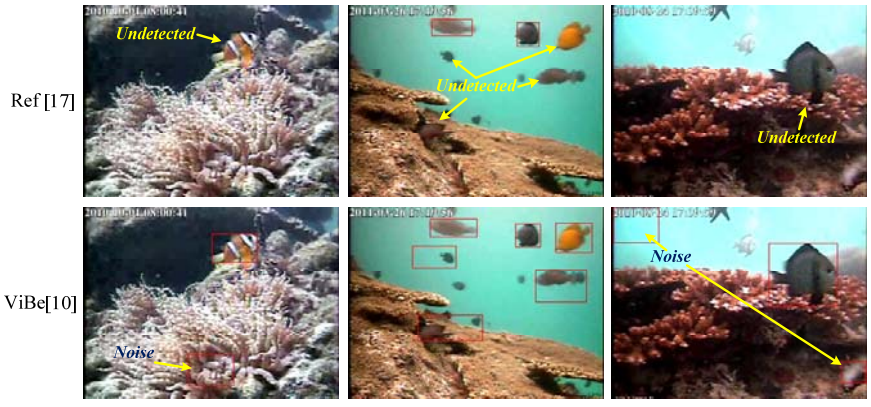


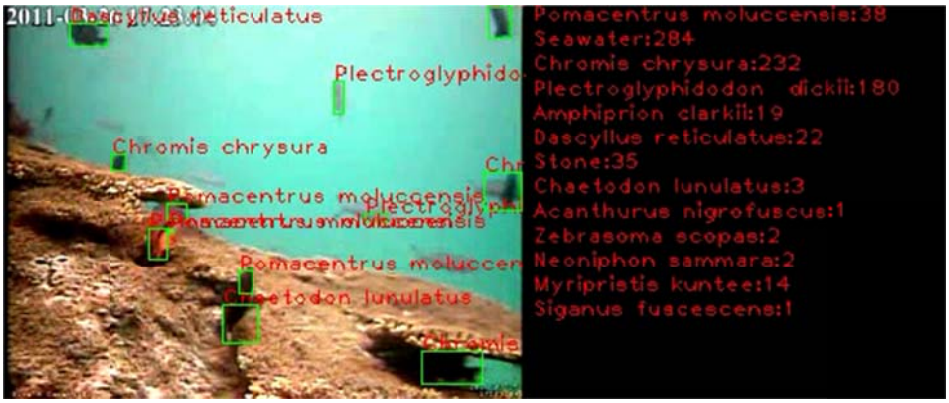
Figure 6. Results of two different fish detection methods

Many state-of-the-art object detection methods have been proposed in recent years. This work does not pay much attention on an object detection algorithm itself. We choose ViBe which has been already reported as an effective tool for underwater object detection [39]. Its main disadvantage is that a lot of false positives will be proposed compared with other methods [16] as shown in Figure 6. However it is

preferred due to its ability of detecting more proposals. To tackle the false positive problem, we add some unwanted patches such as stones, seawater and corals in the training dataset so as to hand over the problem to the following UW-*CNN* classification model. As illustrated in figure 6, these false positive proposals will be classified as noise. The false positive proposals are mainly caused from water waves generated by moving objects and objects are far away from the camera. From table 3, we can also see that all the test images of unwanted examples are classified correctly as there are. Readers can also find some supplement results from Github. (https://github.com/xingkongguye/Underwater).

The proposed framework for underwater object recognition can be implemented for real video monitoring on AUV and underwater observation systems. On the above mentioned experiment environment, our system processes images at 23 frames per second (FPS). As objects moving in the water are quite slow, we also reduce the frame rate by skipping one frame every two frames, in order to meet the real time requirement. Figure 7 shows some results on real-time videos from Fish4Knowledge video dataset. The statistical results on the right side are calculated by the trajectory-based weighted probabilities decision mechanism which makes the results more accurate.

(a) Detection results on 10th second



(b) Detection results on 20th second

Figure 7. Results on real-time Fish4Knowledge videos

Figure 8 shows seven series patches on the trajectory of five different fishes. We can see that, in the freely swimming environment, one fish has significantly different shapes. Figure 8 also gives the weights of every patch and probabilities of the truth

category. We can see that it is impractical to predict the correct label of the object exactly for every patch, especially for the fifth series of patches. It illustrates that small patches usually confuse the recognition system; however the result can be rectified as the object moves closer. With the help of the trajectory based decision mechanism, our framework can better identify the object appeared on series of frames. Taking the fifth series of objects as an example, the first several frames are misclassified due to the same size of the patches and unfavorable angles. However the result will be corrected as the object moving close to the camera. We also capture two trajectories of false positive proposals, i.e., water wave and stone. By playing back the frames of the sixth trajectory, we observed that a fish suddenly disappeared and churned the water. For the seventh trajectory, we can see that there is a fish far away from the camera appeared and disappeared from time to time. And we can see that the classifier can recognition them as noise.

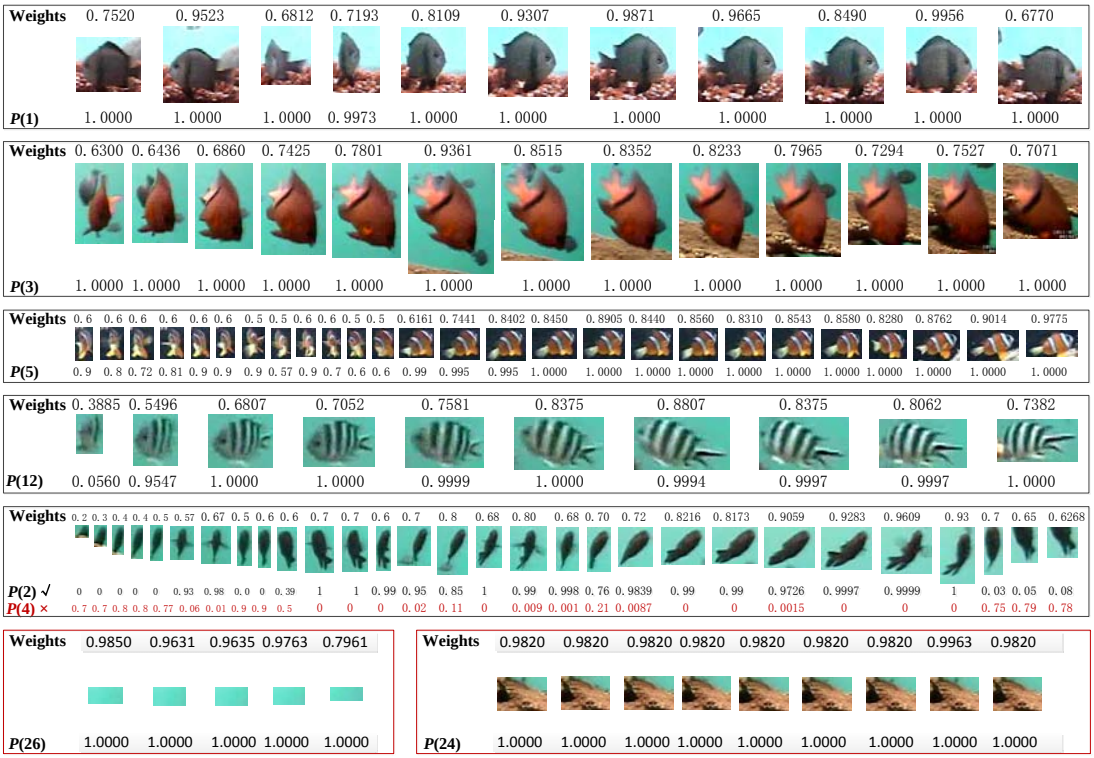


Figure 8. Results for five series patches of different fishes

5. Conclusion

Through ocean observation, we can better understand the ocean environment changes and the behaviors of its resident creatures. With continuous scientific and technological advances, it allows us to explore the ocean in scientific and noninvasive ways, such as underwater video technologies. A plenty of underwater videos are continually collected by autonomous underwater vehicles, underwater robots and video monitoring systems, which give us opportunities to make detailed observations and collect samples of unexplored ecosystems. In performing the ocean observing tasks, the ability of underwater image and video analysis is the key to a success, especially with the low quality videos in low-light and high-noise underwater environments. Considering low contrast caused by the low illumination environment,

this work presented a CNN knowledge transfer framework to extract abstract features from relatively low contrast image, which can perform better than traditional manual features in such a bad situation. To overcome insufficient training set problem, a transfer approach is proposed to learn a deep CNN model for special underwater object recognition, together with the help of data augmentation. Even with the insufficient training set trouble, the transfer approach can well learn a deep CNN model for the special underwater object recognition. We also proposed a weighted probabilities decision mechanism based on the trajectory of a series of frames, in order to better identifying objects from underwater video. This research work can be further applied on autonomous underwater vehicles to automatically identify underwater object in real time.

Acknowledgments

We would like to express our sincere appreciation to the anonymous reviewers for their insightful comments, which have greatly aided us in improving the quality of the paper.

This work is supported by the National Natural Science Foundation of China (No. 61401413, 41576011), China Postdoctoral Science Foundation (No. 2015T80749), Natural Science Foundation of Shandong Province (No. ZR2014FQ023), Open Funding of State Key Laboratory of Applied Optics and NVIDIA Academic Hardware Grant.

References

[1] D. Mallet and D. Pelletier, "Underwater video techniques for observing coastal marine biodiversity: a review of sixty years of publications (1952–2012)," *Fisheries Research,* vol. 154, pp. 44-62, 2014.

[2] D. P. Struthers, A. J. Danylchuk, A. D. Wilson, and S. J. Cooke, "Action cameras: Bringing aquatic and fisheries research into view," *Fisheries,* vol. 40, pp. 502-512, 2015.

[3] M. Cappo, E. Harvey, and M. Shortis, "Counting and measuring fish with baited video techniques-an overview," in *Australian Society for Fish Biology Workshop Proceedings*, 2006, pp. 101-114.

[4] T. Fukuba, T. Miwa, S. Watanabe, N. Mochioka, Y. Yamada, M. J. Miller*, et al.*, "A new drifting underwater camera system for observing spawning Japanese eels in the epipelagic zone along the West Mariana Ridge," *Fisheries Science,* vol. 81, pp. 235-246, 2015.

[5] F. Bonin-Font, G. Oliver, S. Wirth, M. Massot, P. L. Negre, and J.-P. Beltran, "Visual sensing for autonomous underwater exploration and intervention tasks," *Ocean Engineering,* vol. 93, pp. 25-44, 2015.

[6] P. X. Huang, B. J. Boom, and R. B. Fisher, "Hierarchical classification with reject option for live fish recognition," *Machine Vision and Applications,* vol. 26, pp. 89-102, 2015.

[7] M.-C. Chuang, J.-N. Hwang, K. Williams, and R. Towler, "Tracking live fish from low-contrast and low-frame-rate stereo videos," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 25, pp. 167-179, 2015.

[8] M. J. H. Hickford and D. R. Schiel, "Catch vs count: Effects of gill-netting on reef fish populations in southern New Zealand," *Journal of Experimental Marine Biology and Ecology,* vol. 188, pp. 215-232, 5/26/ 1995.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.

[10] D. Pelletier, K. Leleu, G. Mou-Tham, N. Guillemot, and P. Chabanet, "Comparison of visual census and high definition video transects for monitoring coral reef fish assemblages," *Fisheries Research,* vol. 107, pp. 84-93, 2011.

[11] D. T. Jones, C. D. Wilson, A. De Robertis, C. N. Rooper, T. C. Weber, and J. L. Butler, "Evaluation of rockfish abundance in untrawlable habitat: combining acoustic and complementary sampling tools," *Fishery Bulletin,* vol. 110, pp. 332-343, 2012.

[12] A. Klimley and S. Brown, "Stereophotography for the field biologist: measurement of lengths and three-dimensional positions of free-swimming sharks," *Marine Biology,* vol. 74, pp. 175-185, 1983.

[13] J. Lines, R. Tillett, L. Ross, D. Chan, S. Hockaday, and N. McFarlane, "An automatic image-based system for estimating the mass of free-swimming fish," *Computers and Electronics in Agriculture,* vol. 31, pp. 151-168, 2001.

[14] E. Harvey, M. Cappo, M. Shortis, S. Robson, J. Buchanan, and P. Speare, "The accuracy and precision of underwater measurements of length and maximum body depth of southern bluefin tuna (Thunnus maccoyii) with a stereo–video camera system," *Fisheries Research,* vol. 63, pp. 315-326, 2003.

[15] D. Walther, D. R. Edgington, and C. Koch, "Detection and tracking of objects in underwater video," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 2004, pp. I-544-I-549 Vol. 1.

[16] C. Spampinato, Y.-H. Chen-Burger, G. Nadarajan, and R. B. Fisher, "Detecting, Tracking and Counting Fish in Low Quality Unconstrained Underwater Videos," *VISAPP (2),* vol. 2008, pp. 514-519, 2008.

[17] C. Spampinato, D. Giordano, R. Di Salvo, Y.-H. J. Chen-Burger, R. B. Fisher, and G. Nadarajan, "Automatic fish classification for underwater species behavior understanding," in *Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams*, 2010, pp. 45-50.

[18] M.-C. Chuang, J.-N. Hwang, K. Williams, and R. Towler, "Automatic fish segmentation via double local thresholding for trawl-based underwater camera systems," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*, 2011, pp. 3145-3148.

[19] B. J. Boom, J. He, S. Palazzo, P. X. Huang, C. Beyan, H.-M. Chou*, et al.*, "A research tool for long-term and continuous analysis of fish assemblage in coral-reefs using underwater camera footage," *Ecological Informatics,* vol. 23, pp. 83-97, 2014.

[20] D. Lee, G. Kim, D. Kim, H. Myung, and H.-T. Choi, "Vision-based object detection and tracking for autonomous navigation of underwater robots," *Ocean Engineering,* vol. 48, pp. 59-68, 2012.

[21] M.-C. Chuang, J.-N. Hwang, K. Williams, and R. Towler, "Tracking live fish from low-contrast and low-frame-rate stereo videos," *Circuits and Systems for Video Technology, IEEE Transactions on,* vol. 25, pp. 167-179, 2015.

[22] D. Charalampidis, M. Gundam, G. E. Ioup, J. W. Ioup, and C. H. Thompson, "Stereo image segmentation with application in underwater fish detection and tracking," in *SPIE Defense+ Security*, 2015, pp. 94760H-94760H-9.

[23] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE,* vol. 86, pp. 2278-2324, 1998.

[24] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Computer vision–ECCV 2014*, ed: Springer, 2014, pp. 818-833.

[25] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based R-CNNs for fine-grained category detection," in *Computer Vision–ECCV 2014*, ed: Springer, 2014, pp. 834-849.

[26] K. Kang, W. Ouyang, H. Li, and X. Wang, "Object detection from video tubelets with convolutional neural networks," presented at the IEEE Conference on Computer Vision and Pattern Recognition 2016, 2016.

[27] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, "DeepFish: Accurate underwater live fish recognition with a deep architecture," *Neurocomputing,* 2015.

[28] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1717-1724.

[29] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng*, et al.*, "Decaf: A deep convolutional activation feature for generic visual recognition," *arXiv preprint arXiv:1310.1531,* 2013.

[30] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering,* vol. 22, pp. 1345-1359, 2010.

[31] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Advances in Neural Information Processing Systems*, 2014, pp. 3320-3328.

[32] M. Xie, N. Jean, M. Burke, D. Lobell, and S. Ermon, "Transfer Learning from Deep Features for Remote Sensing and Poverty Mapping," *arXiv preprint arXiv:1510.00098,* 2015.

[33] O. Penatti, K. Nogueira, and J. Santos, "Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 44-51.

[34] D. Ciresan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *2012 IEEE Conference on Computer Vision and Pattern Recognition, ,* 2012, pp. 3642-3649.

[35] B. J. Boom, P. X. Huang, J. He, and R. B. Fisher, "Supporting ground-truth annotation of image datasets using clustering," in *2012 21st International Conference on Pattern Recognition (ICPR)*, 2012, pp. 1542-1545.

[36] H. Yu and H. Liu, "Linear regression for head pose analysis," in *Neural Networks (IJCNN), 2014 International Joint Conference on*, 2014, pp. 987-992.

[37] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma*, et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision,* vol. 115, pp. 211-252, 2015.

[38] S. Varadarajan, P. Miller, and H. Zhou, "Region-based Mixture of Gaussians modelling for foreground detection in dynamic scenes," *Pattern Recognition,* vol. 48, pp. 3488-3503, 2015.

[39] R. B. Fisher, Y.-H. Chen-Burger, D. Giordano, L. Hardman, and F.-P. Lin, *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*: Springer, 2016.

[40] G. Nadarajan, Y.-H. Chen-Burger, and R. B. Fisher, "Semantics and Planning Based Workflow Composition for Video Processing," *Journal of grid computing,* vol. 11, pp. 523-551, 2013.

[41] O. Barnich and M. Van Droogenbroeck, "ViBe: A universal background subtraction algorithm

for video sequences," *IEEE Transactions on Image Processing,* vol. 20, pp. 1709-1724, 2011.

[42] T. Bouwmans, "Traditional and recent approaches in background modeling for foreground detection: An overview," *Computer Science Review,* vol. 11, pp. 31-66, 2014.

[43] C. Spampinato, S. Palazzo, D. Giordano, I. Kavasidis, F.-P. Lin, and Y.-T. Lin, "Covariance based Fish Tracking in Real-life Underwater Environment," in *VISAPP (2)*, 2012, pp. 409-414.

[44] B. J. Boom, P. X. Huang, J. He, and R. B. Fisher, "Supporting ground-truth annotation of image datasets using clustering," in *Pattern Recognition (ICPR), 2012 21st International Conference on*, 2012, pp. 1542-1545.

[45] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick*, et al.*, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the ACM International Conference on Multimedia*, 2014, pp. 675-678.

[46] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov*, et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1-9.

[47] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," presented at the International Conference on Learning Representations, 2015.

[48] J. Davis and M. Goadrich, "The relationship between Precision-Recall and ROC curves," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 233-240.